J. COMBIN. MATH. COMBIN. COMPUT. 127a(2025) 4309--4323

Journal of Combinatorial Mathematics and Combinatorial Computing
www.combinatorialpress.com/jcmcc

Combinatorial Press

Research on the Application of Virtual Reality Technology in Cultural Heritage Digitisation

Xiaojian Chen[1], Jiaqi Yuan[2,✉]

[1] *Guangzhou Huashang College, Guangzhou, Guangdong, 511300, China*

[2] *Zhujiang College of South China Agricultural University, Guangzhou, Guangdong, 510900, China*

ABSTRACT
With the continuous development of virtual reality technology, its application in the digitization of cultural heritage has been constantly emphasized and applied, which has an important role and significance for the protection and inheritance of cultural heritage. This paper proposes a rendering algorithm that combines LOD algorithm and occlusion rejection algorithm. The article firstly carries out theoretical research on the relevant theories and rendering processes of LOD algorithm and occlusion removal algorithm, and finally takes the cultural heritage of Shennongjia as the research object to analyze the performance of this paper's algorithm in rendering different landscape scenes of the cultural heritage of Shennongjia. This paper concludes that in the high configuration machine, the algorithm of this paper improves the rendering performance by 587% in the resolution of 1280*720, and improves the rendering performance by 1061% in the resolution of 1920*1080. In the low configuration machine, the algorithm in this paper improves the performance by 653% in 1280*720 resolution and 770% in 1920*1080 resolution. Rendering frame rate LOD combined with occlusion culling algorithm (132.65fps) > occlusion culling algorithm (79.88fps) > LOD method (18.02fps) > without any optimization algorithm (5.32fps). The total number of rendering triangles is without any optimization algorithm (55.65) > LOD algorithm (16.78) > occlusion culling algorithm (3.64) > algorithm of LOD combined with occlusion culling (1.05).

Keywords: virtual reality technology, cultural heritage, LOD algorithm, occlusion rejection algorithm

1. Introduction

With the far-reaching impact of the digital creative industry on people's lives, virtual reality has been vigorously developed in recent years, and there is nothing more highly regarded in the science and technology industry than virtual reality technology [1-3]. Virtual reality technology, is the use of computer simulation to create a three-dimensional space of the virtual environment, so that the

✉ Corresponding author.
E-mail address: 18502020817@163.com (J. Yuan).
Received 01 February 2024; Revised 20 April 2024; Accepted 10 November 2024; Published Online 15 April 2025.
DOI: 10.61091/jcmcc127a-244

experience as if the person is immersed in the virtual environment without restrictions. Virtual reality technology has the advantageous characteristics of multi-sensory, presence, interactivity, autonomy and so on [4-7]. The fields penetrated by virtual reality are extremely wide, such as movie production, game animation, broadcasting and television, medical care, entertainment, military aerospace, marketing display, education, skill training and so on.

Using virtual reality technology, a systematic and in-depth research on the digital protection and dissemination of cultural heritage has been carried out; in order to rescue, protect and develop cultural heritage in a timely manner, a theoretical system of digital protection and dissemination of cultural heritage based on virtual reality technology has been proposed [8-10]. It provides theoretical support and methodological guidance for further developing and building the characteristic cultural industry of cultural heritage and exploring the inheritance and innovation mode of cultural heritage. The use of virtual reality technology has obvious advantages for enhancing sensory experience, innovating content expression, optimizing dissemination mode, and revitalizing inheritance effect, and can become a breakthrough in the creative transformation and innovative development of cultural heritage [11-12]. It has important practical significance and application value for realizing the development goal of rural revitalization strategy, promoting digital protection and revitalization of cultural heritage, creating a high-quality cultural content ecology, and expanding the depth of content dissemination to strengthen the dissemination effect [13-14].

Literature [15] proposes a solution to improve the accessibility and retrieval accuracy of digital cultural heritage resources in Europe through a system that integrates artificial intelligence, natural language technologies, web services and api. Literature [16] challenges the notion that global heritage collections are becoming increasingly visible and accessible. It suggests that future digital analysis of cultural heritage may only be possible through image corpora provided by museums. Literature [17] will reiterate the importance of a stricter ethic of digital practice and make recommendations for the management and representation of historical data so that cultural heritage institutions and digital humanists can better inform users of the lack of and subjectivity that shapes digital datasets and archives. Literature [18] describes a new 3D digitization method that allows for faster and more accurate mapping and faster digital recording underwater. It provides an innovative solution not only for archaeological field research, but also for heritage management and public outreach. Literature [19] analyzes the complexity of growing cultural data and proposes a cultural big database as an effective way of storing and retrieving cultural big data, providing an integrated and high-performance approach to big data analysis of cultural heritage. It outperforms traditional knowledge bases in terms of storage space and storage retrieval time for cultural big data. Literature [20] aims to situate the archival, collecting and curatorial components of past design history magazines within the broader discipline of design history and connect them to scholarship outside of the design history magazines themselves, with great implications for design historians, students and the public's future engagement with design history. Literature [21] proposes the development of an expert system by analyzing the main components of an expert system - the user interface, the database and the knowledge base. This expert system has identified different parameters and proposed a satisfactory solution by analyzing the theoretical methodology of representative tangible cultural heritage. Literature [22] proposes a framework for the purpose of illumination enhancement of heritage images. The obtained images were traded off between local contrast improvement, detail enhancement and maintaining the naturalness of the image. Experiments are conducted on a collected dataset of cultural heritage images and the results show that our proposed approach is able to achieve a wide range of performance objectives.

This paper proposes a scene rendering algorithm that combines the LOD algorithm and the culling algorithm on the basis of the traditional testing algorithm. The article firstly researches the LOD algorithm, theoretically researches the quadtree LOD algorithm based on the rule network model and constructs the modeling of LOD, and then researches the related theory and rendering process of the occlusion culling algorithm. Finally, we take China's natural cultural heritage of Shennongjia as an example for empirical testing, rendering different landscape scenes of Shennongjia, comparing the algorithm proposed in this paper with other traditional algorithms, testing the efficiency of this paper's algorithm on the shadow map rendering time and the screen rendering time, as well as the rendering performance under different resolutions of 1280*720 and 1920*1080.

2. Method

The research content of this paper is the optimization algorithm of large-scale complex scene rendering, through the exploration of related rendering algorithms to accelerate the rendering speed of large-scale scenes of cultural heritage, this section will be the scene accelerated rendering algorithms of the culling algorithm and the LOD algorithm of the two efficiently combined, in order to more efficiently carry out the use of the cultural heritage in the digitization.

2.1. LOD algorithm

2.1.1. Quadtree LOD algorithm based on rule-based network modeling. The quadtree algorithm based on the regular mesh is the process of constantly quadrupling a terrain block. Throughout the operation of the terrain block, it is determined whether or not to quadrate the terrain block by determining whether or not the fineness of the rendering is satisfied. If the rendering conditions are met, then the terrain block is rendered to the screen in the form of triangles. If it does not meet the requirements, then it is necessary to continue to divide it into four blocks to cycle through the judgment until the result that meets the requirements is achieved.

Because in the process of dividing the blocks need to judge whether to meet the conditions of the division, so it will result in different levels of the blocks in different areas. This creates a topographic grid structure map with different levels. The approximate segmentation process of the quadtree is shown in Figure 1.

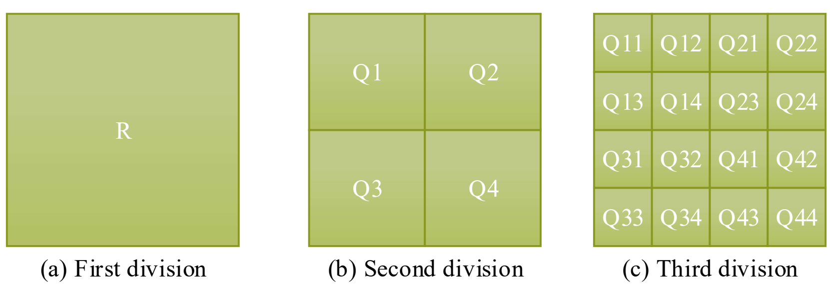

Fig. 1. The four-fork tree is divided into layers

2.1.2. LOD modeling. LOD modeling is to reduce the complexity of image drawing and improve the efficiency of the drawing algorithm by appropriately simplifying the details of the object in different levels while maintaining the visual characteristics of the drawn object. In the process of drawing large-scale complex terrain, the fineness of drawing is determined by judgment factors such as the distance from the viewpoint. In short, it is in the region of close distance to use high fineness resolution to draw, in the region of far distance to choose in the naked eye can not distinguish the difference, the use of lower fineness to draw, so as not to affect the visual characteristics of the entire terrain. At this stage, the more mainstream method is to use the quadtree LOD model, which is more convenient and effective compared with other LOD models.

In quadtree modeling, it is essentially a quadtree layering of the original grid data to construct the corresponding pyramid. For the original elevation map data, starting from the whole map as the root node of the quadtree, i.e., the top of the pyramid, it is recursively divided into four equal parts into four sub-maps, which are constructed as sub-nodes, i.e., the next layer of the pyramid, and then the same operation is carried out for each sub-map until every pixel of the whole map is classified into the leaf nodes. Each layer after processing has twice the resolution and four times the amount of information than the previous layer. An example of quadtree division at different layers is shown in Fig. 2.

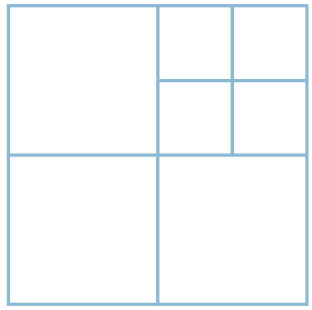

Fig. 2. The partition of the four cross trees

2.1.3. Quadtree hierarchies. Usually when you get a set of elevation map data of terrain, you have to preprocess it. Because the algorithm in this paper is based on the structure of quadtree to deal with the data, so it is necessary to preprocess the elevation map with quadtree. That is to say the resolution must be (2n+1) (2n+1), which makes it possible to recursively quadrature the terrain blocks. In case the elevation map does not meet this resolution, it should be extended accordingly to the required resolution. Filling the region with null values satisfies the data format requirements and does not cause data interference. The paper utilizes a full range elevation map to meet the resolution requirement, and constructs a pyramid model of terrain blocks quickly by building a quadtree, and then combines it with the LOD algorithm to realize the efficient rendering of terrain. Therefore, the terrain blocks are required to be square blocks, which are easy to be used for terrain division. An example of a quadtree division is shown in Figure 3, which demonstrates the division of a quadtree. The original square is the root node. It is then divided into four squares with four child nodes, and then the two squares below are subdivided. Squares with the same area on the plane area are on the same level.

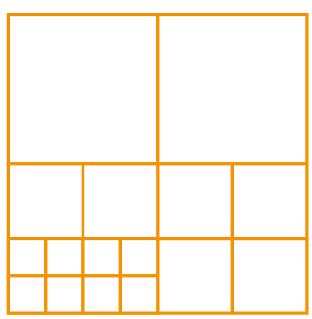

Fig. 3. The partition of the four-fork tree

According to the structure of a quadtree, since the number of nodes in each layer is four times the number of nodes in the previous layer, the information stored is also four times finer than the previous layer. In the drawing process, for the same region, using the nodes at the lower level for drawing will produce a better reduction. This is because in the same region, the nodes at the lower level contain more sampling point data, and naturally more details are plotted. Therefore, the terrain drawn by deeper nodes is also closer to the original terrain. In the mapping algorithm that takes the distance between the viewpoint and the surface roughness as the node evaluation standard, the principle should be followed: in the area closer to the viewpoint, the higher level nodes should be used to draw the details in order to get the clear geomorphic features. On the contrary, distant areas can be drawn with a relatively low level of detail to get the same effect. In addition, the nodes in the upper levels of the quadtree are suitable for depicting relatively flat areas, while areas with severe undulations should be described by nodes in the lower levels to ensure that the surface characteristics of the terrain are restored. Before rendering, the data of the quadtree structure is evaluated from top to bottom nodes. Find the children of the nodes sequentially if the refinement division criteria are met. Until the dissatisfaction of the division conditions or the leaf nodes are reached, then these nodes are processed for drawing. The inverted tree of the quadtree is shown in Figure 4.

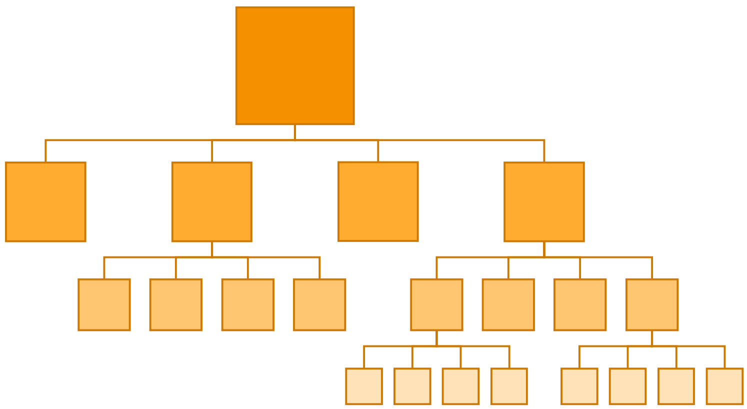

Fig. 4. Inverted tree

2.2. Occlusion Rejection Algorithm

The Portal system generally pre-processes the scene, either manually or automatically. The scene will be divided into different regions, e.g. into rooms, corridors, etc. Those doors and windows neighboring different regions will be recognized as portals. Doors and windows that are adjacent to different regions are identified as portals. The graph that describes the topology of the scene, with regions as graph nodes and portals as neighboring edges, is called Cell-and-Portal-Graph (CPG). At real-time runtime, the CPG graph will be utilized to traverse the Portals in the region, crop these Portals using view cones, and then use the cropped view cones to do view cone collisions on objects in the scene. The scene-to-scene CPG is shown in Fig. 5, where it can be seen that each node in the CPG corresponds to a region, and each edge corresponds to a Portal.

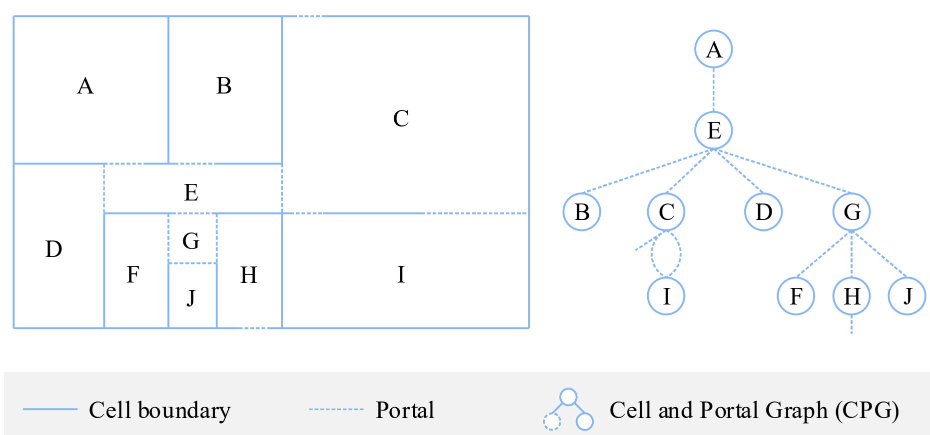

Fig. 5. The CPG scenario and scenario

The Portal system determines the visibility of a region through viewcone cropping as shown in Figure 6, which illustrates the process of determining the visible region through viewcone cropping in this scene. For the dynamic objects placed in the scene, you can use the cropped view cone to do collision test on them to determine the visibility.

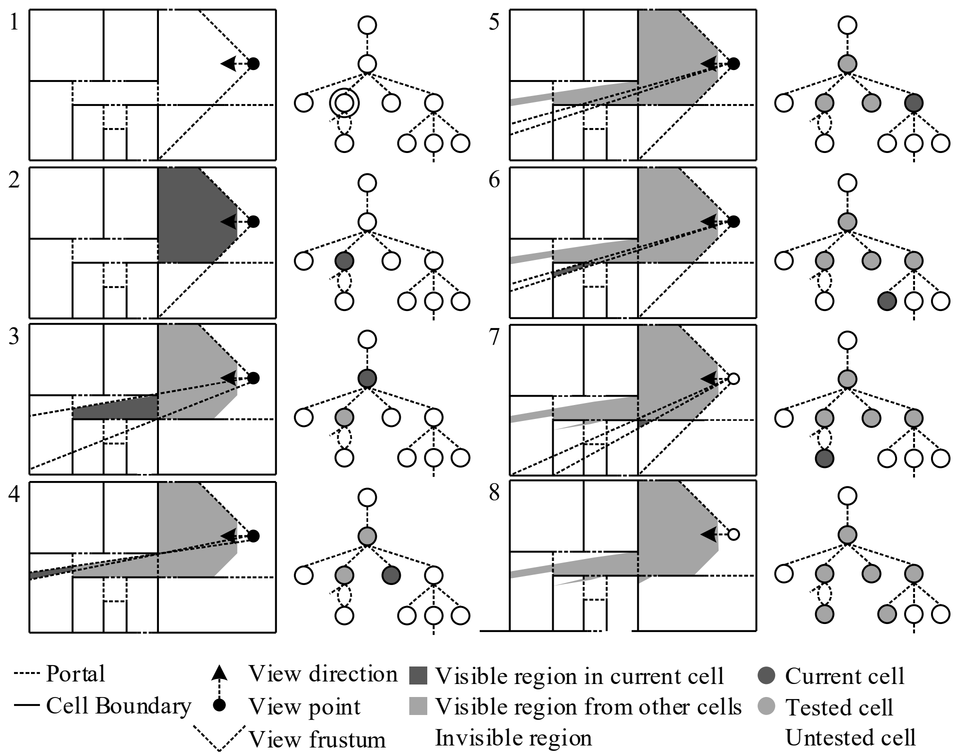

Fig. 6. The portal system is visible by the visual cone

2.3. *Simplifying the Rendering Equation*

There is a more intuitive form of lighting understanding in real-time rendering that considers whether incident light in a certain direction is visible at the currently shaded pixel point, and adds a visibility term to the calculation:

$$L_o(p,\omega_o)=\int_{\omega^+} L_i(p,\omega_i)f_r(p,\omega_i,\omega_o)\cos\theta_i V(p,\omega_i)d\omega_i \tag{1}$$

where $L_o(p,\omega_o)$ is the outgoing light, $L_i(p,\omega_i)$ is the incident light, $f_r(p,\omega_i,\omega_o)$ is the BRDF with clamping weights, and $V(p,\omega_i)$ represents the visibility.

The ambient light terms added from the reflection equation to the rendering equation and in the classical Blinn-Phong lighting model are approximated as constants. In screen space, since the raw geometric information of the scene has already been processed, leaving only the depth, normals, etc. recorded in two dimensions, and since the overhead of many methods lies in the fact that the process of finding the incident light leads to a large number of additional computations, it is useful to assume constant indirect illumination, modeled after constant ambient light, i.e., to make L_i a certain value.

Furthermore, consider the nature of the integral. Usually the integral of a product is not equal to the product of integrals, but an approximation exists. Consider the L_i and BRDF terms of the rendering equation, when the BRDF is high light, it has a small range of regions in the outgoing direction, i.e., the contribution of this incident light to the coloring result is small. When the BRDF is diffuse, it reflects in all directions, but the transition between different directions is smoother. That is, when the BRDF is highlight or diffuse reflection, respectively, satisfy one of the conditions under which Eq. approximately holds:

$$\int_{\Omega} f(x)g(x)dx \approx \frac{\int_{\Omega_G} f(x)dx}{\int_{\Omega_G} dx} \cdot \int_{\Omega} g(x)dx \tag{2}$$

This method focuses on the case where the BRDF is a diffuse reflection-like material, which meets the conditions for this approximation to hold.

At this point, the rendering equation can be simplified by presenting the 1 term and the BRDF term in Eq. L_i from within the integral and knowing that the BRDF of the diffuse reflective material is the constant $\frac{\rho}{\pi}$:

$$L_o \approx \frac{\rho}{\pi} \cdot L_i \cdot \int_{\Omega^+} V \cos\theta_i d\omega_i \tag{3}$$

The remaining variable to be calculated then is the visibility V term of whether or not one can be affected by indirect light from this constant at different coloring points.

2.4. Use of multiplexed data and algorithmic steps

For the depth information map data rendered separately from the dynamic and static object lists, the amount to be compared in the algorithm should be compared to the smaller value sampled from the two depth maps for subsequent use.

The depth information data is stored in a buffer in the rendering pipeline, and each depth value corresponds to a screen pixel coordinate point, which can be represented by a two-dimensional matrix. Let the size of the buffer be $m \times n$. The depth information generated before using this method is:

$$D_{origin} = \begin{pmatrix} Do_{1,1} & \dots & Do_{1,n} \\ \dots & \dots & \dots \\ Do_{m,1} & \dots & Do_{m,n} \end{pmatrix} \tag{4}$$

Instead, two copies of the depth information data are saved after using this method, one is generated for the static object list Q_s, noted as:

$$D_{static} = \begin{pmatrix} Ds_{1,1} & \dots & Ds_{1,n} \\ \dots & \dots & \dots \\ Ds_{m,1} & \dots & Ds_{m,n} \end{pmatrix} \tag{5}$$

Another copy is generated for the current dynamic object list Q_d, denoted:

$$D_{dynamic} = \begin{pmatrix} Dd_{1,1} & \dots & Dd_{1,n} \\ \dots & \dots & \dots \\ Dd_{m,1} & \dots & Dd_{m,n} \end{pmatrix} \tag{6}$$

Then set the screen coordinate of the currently calculated coloring point to (x, y) and its depth value to d.

Since the purpose of detecting depth is all about confirming whether the current coloring point is occluded by other elements, i.e., checking $d \le D_o x, y$. Therefore, after applying this method, it should be modified to take a small amount of the corresponding position of the two copies of the depth data in order to confirm that neither the static element nor the dynamic element that is in that position is occluding the current coloring point, i.e.:

$$d \le \min\left(Ds_{x,y}, Dd_{x,y}\right) \tag{7}$$

The display part of the whole presentation system is divided into two: the display scene and the roaming form (window or viewport). The display part draws the background console for information. Forms implementation, we use the library as a very small window system interface to complete. The information displayed in the background console mainly includes: the type of rendering technique currently used, the number of draw calls, the tilt refresh rate, and the number of triangular facets, in order to do an accurate analysis of the experimental results later. The design of the rendering process is shown in Figure 7.

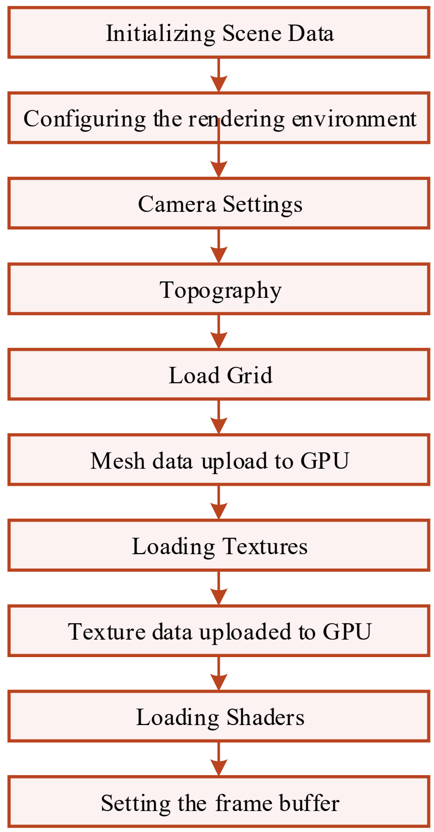

Fig. 7. Rendering process design

3. Testing and application of rendering algorithms in the digitization of cultural heritage

This experiment takes the large-scale landscape scene of Shennongjia cultural heritage as an example, and adopts two comparisons, horizontal comparison and vertical comparison, to analyze the efficiency of the algorithm. In the experiment, this experiment mainly measures two variables, frameRate, the frame rate of scene rendering, and triCount, the number of triangles rendered. The author measures these two results in the following four cases: first, the occlusion culling algorithm and the LOD algorithm are not applied at all. Second, only the LOD algorithm is used. Three, only the occlusion culling algorithm is used. Four, the LOD algorithm is combined with the occlusion culling algorithm. Compare the rendering performance of the proposed algorithm with that of the taskless optimization algorithm, and compare how much the rendering speed of the proposed algorithm is improved and how much the rendering time is saved in different machine configurations.

3.1. Rendering Performance Testing

The comparison of rendering performance under different configuration machines is shown in Table 1 (in frames per second). After comparison, in the high configuration machine, the algorithm in this paper, compared with the forced light step scattering integral calculation method per pixel, compared with not applying any optimization algorithm, in the resolution of 1280*720, the performance is improved by 587%, and in the resolution of 1920*1080, the performance is improved by 1061%. As you can see, modern GPU computing power has been very powerful, even with the most primitive algorithms, you can still get smoother results. However, in real-world applications (such as large-scale 3D games), the GPU needs to use a lot of computation for rendering other content in the application, so optimization of the algorithms is still necessary. Next, we did a test on a low-configuration machine2 to observe and discuss the results of the test. We also tested the rendering performance of this paper's algorithm in 1280*720 and 1920*1080 resolutions, respectively, and compared the performance of this paper's algorithm to the forced per-pixel light-stepping scattering integral calculation method in the higher-configuration machine, the performance of this paper's algorithm was improved by 653% in the resolution of 1280*720, and by 7% in the resolution of 1920*1080, as opposed to the method of calculating the per-pixel forced light-stepping scattering integral calculation, and by 7% in the resolution of 1920*1080. At 1280*720 resolution, the performance is improved by 653%, and at 1920*1080 resolution, the performance is improved by 770%. It can be seen that it is not feasible to compute the scattering integrals directly without any optimization on a low-configuration machine, and the frame rate is already in the low single digits in both resolutions, which is unacceptable in real-time rendering. After optimization of our algorithm, the rendering frame rate can reach 45 frames per second in 1280*720 resolution, which is basically able to meet the demand of real-time rendering.

It is worth noting that the low configuration machine shows better performance than the high configuration machine, which is inconsistent with the traditional cognition. This may be because the algorithm is not very dependent on the hardware, and the high configuration machine has a problem of calculating waiting in the operation.

Table 1. Different configuration machines can compare the color of the device

The discharge color of the high configuration machine can be compared		
Resolution	1280*720	1920*1080
No LoD or Culling	11	10
LOD+Occlusion Culling	98	88
The discharge color of the low configuration machine can be compared		
No LoD or Culling	8	6
LOD+Occlusion Culling	45	38

The experimental efficiencies of the self-shadowing algorithms under seven different scene settings are shown in Table 2. Among them, the shadow map rendering time is the time spent by each self-shadowing algorithm in constructing the relevant shadow map, and the screen rendering time is the time spent in sampling the shadow map to realize particle rendering in the screen space. As can be seen from the comparison of the individual data, the depth opacity shadow map has a shorter shadow map rendering time because it uses only four color channels of a texture map and thus stores four shadow maps. Although increasing the number of shadow map layers optimizes the effect of the algorithm, it does not solve the problem of mutation artifacts that exist when the algorithm renders smoke. In the rendering of seven scenes, the lowest rendering time of the algorithm in this paper is only 10.2ms, while the lowest rendering time of the scene rendering without any algorithm is 14.4ms. Comprehensively, the algorithm proposed in this paper is much lower than that without any optimization algorithm in terms of the rendering time of the shadow map and the rendering time of the screen, which greatly enhances the rendering efficiency, and the self-shadowing effect is the best among the algorithms. The self-shadowing effect obtained is the best among the algorithms, and it also has more stable effect performance under the verification of different transparency.

Table 2. Different algorithm rendering time contrast

Algorithm rendering time			
	Total rendering time(ms)	Shadow rendering time(ms)	Screen rendering time(ms)
Scene 1	12.31	3.51	10.28
Scene 2	10.20	3.22	11.68
Scene 3	11.19	4.23	11.3
Scene 4	12.12	3.28	10.23
Scene 5	10.62	4.15	10.24
Scene 6	11.53	4.66	10.16
Scene 7	11.37	3.51	11.45
No algorithm rendering time			
	Total rendering time(ms)	Shadow rendering time(ms)	Screen rendering time(ms)
Scene 1	15.58	5.23	13.56
Scene 2	15.53	5.25	13.47
Scene 3	15.11	5.63	14.42
Scene 4	14.52	5.29	13.17
Scene 5	14.46	6.34	14.17
Scene 6	15.05	5.21	14.23
Scene 7	14.4	5.44	13.06

3.2. Comparison of measurement efficiency

In this section, four different algorithms are applied to render the scene of the localized terrain of Shennongjia in order to compare the accuracy of the frame rate test under different algorithms. The frame rate comparison and the total number of triangles are shown in Fig. 8. The scene rendering frame rate of LOD combined with the occlusion culling algorithm is the highest (132.65fps), which can significantly improve the rendering efficiency of the scene. The rendering frame rate of the scene using only the occlusion culling algorithm is 79.88fps, and the rendering frame rate of the scene using only the LOD method is 18.02fps, while the rendering frame rate of the scene without any optimization algorithm is only 5.32fps. From this, it can be concluded that the rendering frame rate of the scene obtained by the LOD combined with the occlusion culling algorithm is about 17 times higher than that of the scene rendering without any optimization algorithm, and nearly 17 times higher than that of the scene rendering without any optimization algorithm, and nearly 17 times higher than that of the scene rendering without any optimization algorithm. It is about 17 times higher than the scene rendering frame rate without any optimization algorithm, about 7 times higher than the scene rendering frame rate with only LOD algorithm, and about 1.5 times higher than the scene rendering frame rate with only occlusion culling algorithm. The occlusion culling algorithm is significantly faster than the LOD method in improving the scene rendering speed.

The total number of rendering triangles is an important factor in rendering picture quality, so this experiment applies four different algorithms to scene rendering of the local terrain of Shennongjia, and the author applies the algorithms to a more complex forest scene, which is composed of 1,000 trees, and each tree is represented by 49,207 triangles in the longitudinal comparison. Its scene complexity is up to 49207000 triangles. To compare the total number of triangles rendered by different algorithms, the lowest total number of triangles rendered by the LOD combined with occlusion culling algorithm is 1.05, the total number of triangles rendered by the occlusion culling algorithm is 3.64, the total number of triangles rendered by the LOD algorithm is 16.78, and the total number of triangles rendered by the LOD algorithm without any optimization algorithm is as high as 55.65. The use of LOD combined with occlusion culling reduces the number of unnecessary triangles by Nearly 97% of unnecessary triangles are rendered. The LOD only algorithm reduces 70% of unnecessary triangles and the occlusion culling algorithm reduces 95% of unnecessary triangles. From the experimental data, it can be concluded that the LOD algorithm combined with the occlusion culling algorithm can significantly improve the rendering speed of large-scale complex scenes and reduce a large number of triangles that do not contribute to the final image.

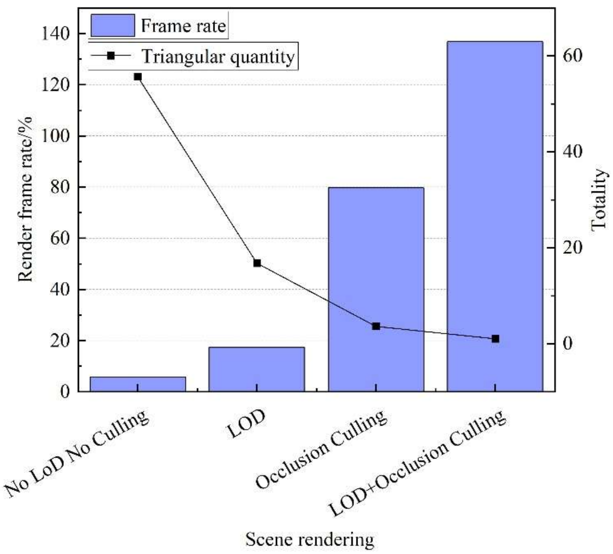

Fig. 8. Frame rate and triangular total comparison

At the end of the experiment, only two methods are used to compare the six scene renderings, which are the method without LOD algorithm and occlusion culling algorithm and the method with LOD combined with occlusion culling algorithm, i.e., the algorithm proposed in this paper. The comparison of the frame rate of the rendering of the six different scenes is shown in Fig. 9. From the comparison graph, it can be seen that the method of LOD combined with occlusion culling algorithm has a substantial improvement in the frame rendering rate than the algorithm without any optimization. The rendering efficiency of scene 4 with LOD combined with occlusion culling algorithm is up to 85.34%, while the rendering efficiency of scene 5 without any optimization algorithm is only up to 8.97%. In terms of culling efficiency, the LOD combined with occlusion culling algorithm approach uses only 3% of the triangles to render the scene realistically. The longitudinal comparison applies the algorithm to more complex scenes, measures their scene acceleration ratios, and compares them with the acceleration ratios of the scenes used for the horizontal comparison. In the more complex scenes, the acceleration effect of the algorithm is more obvious. The experiment proves that the large-scale complex scene rendering optimization algorithm proposed in this paper can improve the rendering efficiency of large-scale complex scenes to a large extent. From the above two data, it can be seen that the algorithm in the more complex scene, the more it can play its acceleration efficiency, the more obvious the acceleration effect on scene rendering.

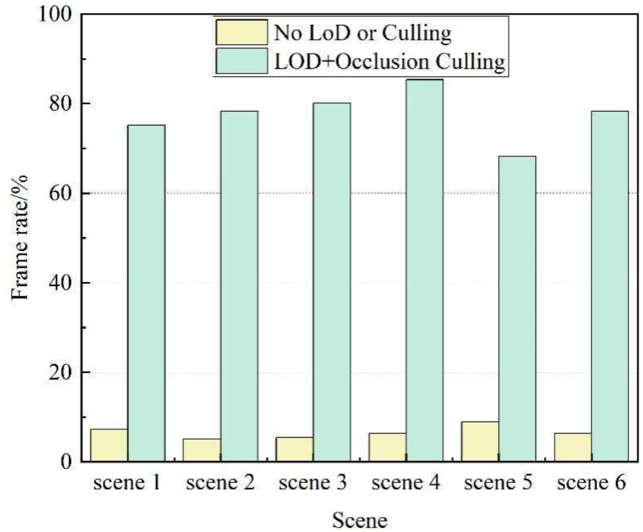

Fig. 9. Scenario rendering frame rate contrast

4. Conclusion

This paper combines both LOD algorithm and occlusion rejection algorithm on the traditional algorithm, and draws the following conclusions.

1) In the high configuration machine, the algorithm in this paper improves the rendering performance by 587% in 1280*720 resolution and 1061% in 1920*1080 resolution. In the low configuration machine, the performance is improved by 653% in 1280*720 resolution and 770% in 1920*1080 resolution.

2) The rendering frame rates of the four algorithms compared are, from highest to lowest, LOD combined with occlusion culling algorithm (132.65fps) > occlusion culling algorithm (79.88fps) > LOD method (18.02fps) > without any optimization algorithm (5.32fps).

3) The total number of rendering triangles for the comparison of the four algorithms, from highest to lowest, is without any optimization algorithm (55.65) > LOD algorithm (16.78) > occlusion culling algorithm (3.64) > algorithm of LOD combined with occlusion culling (1.05).

The virtual scene reconstruction algorithm proposed in this paper is proved to be efficient and feasible to complete the digital reconstruction of cultural heritage. The method of virtual reconstruction not only realizes the effective preservation of cultural heritage, but also promotes the cultural experience.

Fund Project

1. Guangdong Philosophy and Social Science Planning 2024 Discipline Co-construction Project "Study on the Digital Design and Communication of the Brand Image of Guangdong, Hong Kong and Macao Greater Bay Area. No.GD24XYS028

2. This paper was supported by the campus project support of Guangzhou Huashang College Project No. 2023HSDS11.

References

[1] Shallcross, B. . (2017). Materiality and logosphere of the everyday. Teksty Drugie, 2017(1), 334-348.

[2] Nadim, Akhtar, Khan, S., M., & Shafi, et al. (2018). Digitization of cultural heritage: global initiatives, opportunities and challenges. Journal of Cases on Information Technology.

[3] Ponciano, J. J. , Prudhomme, C. , & Boochs, F. . (2021). From Acquisition to Presentation—The Potential of Semantics to Support the Safeguard of Cultural Heritage. Remote Sensing, 13(11), 2226.

[4] Palumbo, R. , Ciasullo, M. V. , Pellegrini, M. M. , Caputo, A. , & Turco, M. . (2022). Locally focused and digitally oriented: examining eco-museums' digitization in a service quality management perspective. The TQM Journal(3), 34.

[5] Bhowmik, N. , Gouet-Brunet, V. , Bloch, G. , & Besson, S. . (2017). Combination of image descriptors for the exploration of cultural photographic collections. Journal of Electronic Imaging, 26(1), 011019.1-011019.13.

[6] Zhang, J. , Zhou, Y. , Xia, K. , Jiang, Y. , & Liu, Y. . (2020). A novel automatic image segmentation method for chinese literati paintings using multi-view fuzzy clustering technology. Multimedia Systems, 26(1), 37-51.

[7] Fallatah, A. . (2021). As-textured as-built bim using sensor fusion, zee ain historical village as a case study. Remote Sensing, 13.

[8] Fanea-Ivanovici, M. , & Pana, M. C. . (2020). From culture to smart culture. how digital transformations enhance citizens' well-being through better cultural accessibility and inclusion. IEEE Access, PP(99), 1-1.

[9] Amelio, & Zarri. (2019). Conceptual encoding and advanced management of leonardo da vinci's mona lisa: preliminary results. Information (Switzerland), 10(10), 321.

[10] Sze - Teng Liong, Yen - Chang Huang, Li, S. , Huang, Z. , Ma, J. , & Gan, Y. S. . (2020). Automatic traditional chinese painting classification: a benchmarking analysis. Computational Intelligence, 36(3).

[11] Younes, G. , Asmar, D. , Elhajj, I. , & Al-Harithy, H. . (2017). Pose tracking for augmented reality applications in outdoor archaeological sites. Journal of electronic imaging, 26(1), 011004.1-011004.12.

[12] Apollonio, F. , Fantini, F. , Garagnani, S. , & Gaiani, M. . (2021). A photogrammetry-based workflow for the accurate 3d construction and visualization of museums assets. Remote Sensing.

[13] Jiao, P. , Li, L. , Yamaguchi, H. , & Tanaka, S. . (2021). Integrated high-definition visualization of digital archives for borobudur temple. Remote Sensing.

[14] Moyano, J. , Nieto, J. E. , José David Bienvenido Huertas, & David Marín. (2020). Validation of close-range photogrammetry for architectural and archaeological heritage: analysis of point density and 3d mesh geometry. Remote Sensing, 12(21), 3571.

[15] Machidon, O. M. , Tavar, A. , Gams, M. , & Dugulean, M. . (2019). Culturalerica: a conversational agent improving the exploration of european cultural heritage. Journal of Cultural Heritage, 41.

[16] Kizhner, I. , Terras, M. , Rumyantsev, M. , Sycheva, K. , & Rudov, I. . (2018). Accessing russian culture online: the scope of digitization in museums across russia. Digital Scholarship in the Humanities.

[17] Alexandra, O. B. , & Julianne, N. . (2021). Encoding the haunting of an object catalogue: on the potential of digital technologies to perpetuate or subvert the silence and bias of the early-modern archive1. Digital Scholarship in the Humanities(3), 3.

[18] Repola, L. , Nicola, S. D. C. , Signoretti, D. , & Leidwanger, J. . (2018). Virtual simulation of a late antique shipwreck at marzamemi, sicily: integrated processes for 3d documentation, analysis and representation of underwater archaeological data. Archaeological Prospection.

[19] Seemu, S. , & Seema, B. . (2019). Cbdr: an efficient storage repository for cultural big data. Digital Scholarship in the Humanities(4), 4.

[20] Zo, H. . (2019). Archives, collections and curatorship: virtual special issue for the journal of design history, 2019. Journal of Design History.

[21] Budak, I. , Santosi, Z. , Stojakovic, V. , Crkvenjakov, D. K. , Obradovic, R. , & Milosevic, M. , et al. (2019). Development of expert system for the selection of 3d digitization method in tangible cultural heritage. Tehnicki Vjesnik(3).

[22] Jmal, M. , Souidene, W. , & Attia, R. . (2017). Efficient cultural heritage image restoration with nonuniform illumination enhancement. Journal of Electronic Imaging, 26(1), 011020.